 jnovic printing images

b22e6ed 10 hours ago

1 contributor

1.54 MB

In [230]: %matplotlib notebook

```
In [131]: import pandas as pd
import csv
import numpy as np
import matplotlib.pyplot as plt
import requests
from pprint import pprint
import scipy.stats as stats

import os
from census import Census
from pprint import pprint
# Census API Key
from config import api_key

buz_info = 44021
zipcode = pd.read_csv("Resources/Public_ZipCode.csv")
zipcode2 = pd.read_csv("Resources/charter_df.csv")
zipcode.head()
```

Out[131]:

|   | Building Name             | District Name        | County | City, State, Zip Code      | Performance Index Score 2015-16 | City       |
|---|---------------------------|----------------------|--------|----------------------------|---------------------------------|------------|
| 0 | Ada Elementary School     | Ada Exempted Village | Hardin | Ada, OH, 45810-1013        | 96.125                          | Ada        |
| 1 | Ada High School           | Ada Exempted Village | Hardin | Ada, OH, 45810-1013        | 91.667                          | Ada        |
| 2 | Sandusky Middle School    | Sandusky City        | Erie   | Sandusky, OH, 44870-2616   | 62.772                          | Sandusky   |
| 3 | Meigs Primary School      | Meigs Local          | Meigs  | Middleport, OH, 45760-9717 | NC                              | Middleport |
| 4 | Meigs Intermediate School | Meigs Local          | Meigs  | Middleport, OH, 45760-9717 | 70.394                          | Middleport |

```
In [132]: #API CALL
#zipcode[["zipcode","extra"]] = zipcode["Zip Code"].str.split("-",expand=True)
zcode = zipcode['Zip Code']
zip_pd = pd.DataFrame(zcode)

zcode2 = zipcode2['Zip Code']
zip_pd2 = pd.DataFrame(zcode2)

zip_pd2.head()
```

Out[132]:

|   | Zip Code |
|---|----------|
| 0 | 45404    |
| 1 | 45207    |
| 2 | 43537    |
| 3 | 44115    |
| 4 | 44663    |

In [133]: zip\_pd.replace('^(\s+|\s+\$)', '', regex=True, inplace=True)

In [134]: zcode.head()

Out[134]:

```
0    45810
1    45810
2    44870
3    45760
4    45760
Name: Zip Code, dtype: int64
```

In [135]: list(zcode)

Out[135]:

```
[45810,
 45810,
 44870,
 45760,
 45760,
 45013,
 43802,
 43307]
```

4320/,  
45656,  
45656,  
45369,  
45503,  
45404,  
45207,  
45631,  
45669,  
43011,  
43901,  
45628,  
43537,  
44122,  
45601,  
44121,  
43604,  
44115,  
44663,  
43344,  
45710,  
44133,  
44857,  
44644,  
43207,  
45414,  
44320,  
44709,  
43604,  
44130,  
44502,  
43609,  
44484,  
45237,  
44883,  
45710,  
44134,  
44111,  
44102,  
44134,  
44134,  
43613,  
44503,  
44817,  
44055,  
45859,  
44256,  
43302,  
43302,  
43430,  
44306,  
44311,  
45850,  
44255,  
44255,  
43138,  
43138,  
43138,  
44030,  
44030,  
44906,  
45601,  
44805,  
44707,  
43920,  
45640,  
43219,  
45850,  
45144,  
45144,  
45651,  
44601,  
45801,  
45212,  
45804,  
44132,  
44102,  
45036,  
45036,

43229,  
45101,  
43119,  
43227,  
45505,  
43229,  
43207,  
45155,  
44555,  
44134,  
43102,  
44111,  
44123,  
43615,  
45044,  
43229,  
43205,  
43232,  
43204,  
45214,  
44112,  
43102,  
44102,  
45202,  
45410,  
45102,  
45103,  
45224,  
44052,  
43207,  
45103,  
45801,  
43207,  
44485,  
45711,  
45415,  
44509,  
44077,  
43614,  
45044,  
44646,  
44270,  
45236,  
44052,  
44060,  
43231,  
45255,  
44804,  
44011,  
44003,  
43004,  
44120,  
44120,  
44647,  
44120,  
45302,  
45302,  
45303,  
43604,  
45303,  
43212,  
43212,  
45005,  
45237,  
45415,  
43571,  
43615,  
44104,  
43701,  
45813,  
44109,  
44052,  
44052,  
43612,  
45813,  
44606,  
44509,  
44103,  
44702

44076,  
43228,  
44212,  
43068,  
43068,  
43064,  
45415,  
44804,  
43035,  
44804,  
45304,  
43315,  
45304,  
43502,  
44114,  
43502,  
43232,  
43055,  
44444,  
43607,  
43204,  
45231,  
43609,  
45814,  
45814,  
44321,  
44135,  
43232,  
44805,  
44805,  
43003,  
44004,  
43103,  
45780,  
43420,  
44807,  
44807,  
44201,  
44875,  
44023,  
45873,  
44615,  
44202,  
44202,  
44010,  
44515,  
43026,  
44011,  
44011,  
44011,  
44012,  
43222,  
44708,  
43512,  
43512,  
45067,  
44428,  
45612,  
44473,  
44305,  
44203,  
45784,  
43713,  
43713,  
44054,  
43221,  
44430,  
44145,  
43452,  
45103,  
45103,  
44333,  
45801,  
45801,  
45801,  
45449,  
44140,  
44140,

44481,  
44122,  
43026,  
43716,  
43716,  
45342,  
43203,  
43224,  
43920,  
43920,  
45434,  
45419,  
44146,  
43213,  
45218,  
44704,  
43906,  
45305,  
45305,  
43311,  
45416,  
43311,  
44811,  
44811,  
44813,  
45420,  
44460,  
45714,  
44118,  
44109,  
44021,  
44401,  
44610,  
44814,  
43155,  
43155,  
45036,  
43209,  
45371,  
45106,  
45371,  
45106,  
44310,  
44305,  
43614,  
45715,  
45431,  
43209,  
45614,  
43074,  
43204,  
43605,  
44087,  
44880,  
45418,  
45107,  
45107,  
43136,  
45682,  
43112,  
45067,  
44450,  
43160,  
44505,  
45236,  
45817,  
45817,  
45895,  
44512,  
44512,  
44223,  
44612,  
44106,  
45237,  
44264,  
45306,  
45306,  
44120,  
44118.

-----,  
44695,  
43402,  
43402,  
43614,  
44430,  
45308,  
45308,  
44124,  
45244,  
45404,  
44147,  
44147,  
45231,  
44060,  
43912,  
45211,  
43913,  
44240,  
44907,  
44402,  
44402,  
43026,  
43209,  
45371,  
44403,  
44403,  
44144,  
44142,  
44144,  
43235,  
44054,  
44142,  
45309,  
43026,  
44266,  
45320,  
44212,  
44044,  
44124,  
43506,  
43506,  
44122,  
44320,  
44854,  
44077,  
44460,  
44256,  
43207,  
43015,  
45681,  
44820,  
44842,  
43952,  
43723,  
44109,  
44004,  
44514,  
45680,  
43567,  
43204,  
43609,  
44021,  
44822,  
45304,  
45377,  
43723,  
44122,  
45233,  
44212,  
43907,  
45658,  
43920,  
43724,  
43724,  
43314,  
44112,  
43725,  
43725,  
-----

45311,  
44406,  
45638,  
44241,  
43064,  
44614,  
43110,  
44406,  
44118,  
44707,  
44413,  
44062,  
44062,  
43315,  
43315,  
43316,  
45005,  
45005,  
45005,  
45408,  
43112,  
44615,  
44615,  
45238,  
44146,  
44103,  
44303,  
43209,  
44710,  
43130,  
44077,  
45314,  
45314,  
43207,  
45822,  
45822,  
44444,  
44081,  
43011,  
43011,  
45459,  
43560,  
43140,  
45040,  
44663,  
43545,  
45042,  
44890,  
45692,  
45014,  
44904,  
44240,  
44146,  
44104,  
45801,  
45840,  
44281,  
45215,  
45601,  
44022,  
44022,  
44470,  
44087,  
44112,  
44483,  
44483,  
43203,  
44483,  
44509,  
43230,  
44024,  
44024,  
44109,  
44105,  
44721,  
44128,  
45248,  
44617,  
43611,



45223,  
43081,  
43608,  
43558,  
43055,  
45619,  
44070,  
44610,  
45211,  
45601,  
44141,  
44230,  
43113,  
43113,  
44135,  
44708,  
43207,  
44102,  
43616,  
45662,  
44683,  
44683,  
44813,  
45066,  
43102,  
45066,  
44720,  
44052,  
43055,  
45103,  
45420,  
45013,  
43202,  
44614,  
45113,  
45245,  
44254,  
45239,  
43410,  
43410,  
45692,  
45417,  
45828,  
43214,  
45251,  
45251,  
45251,  
45224,  
44110,  
43316,  
44827,  
43085,  
43050,  
44028,  
44028,  
44883,  
44408,  
44146,  
44004,  
45830,  
45830,  
43224,  
45373,  
43811,  
44287,  
43402,  
44030,  
44695,  
45831,  
45831,  
44319,  
45373,  
45723,  
44321,  
44321,  
44028,  
44041,  
45881,  
45011

43814,  
43731,  
45238,  
45318,  
45318,  
45385,  
43616,  
43221,  
45044,  
44827,  
44217,  
44212,  
44408,  
45832,  
44805,  
45832,  
44035,  
44255,  
45806,  
43402,  
43528,  
43420,  
43731,  
45030,  
43614,  
44320,  
44410,  
44403,  
44221,  
44125,  
44125,  
44817,  
44506,  
44104,  
44618,  
43214,  
44053,  
44618,  
43050,  
43222,  
43440,  
43440,  
45107,  
44106,  
44053,  
43014,  
43014,  
43123,  
45238,  
43040,  
44432,  
43015,  
43035,  
43968,  
44833,  
45638,  
45638,  
43062,  
45236,  
44685,  
43017,  
43620,  
44117,  
43220,  
43207,  
45638,  
44105,  
43125,  
45066,  
43512,  
45458,  
45801,  
45417,  
43512,  
44721,  
44052,  
45238,  
44620,  
43515,

43515,  
43515,  
45377,  
44109,  
44134,  
45177,  
43524,  
43613,  
43229,  
44104,  
43701,  
45345,  
44408,  
45320,  
44514,  
43214,  
45840,  
45319,  
44139,  
43617,  
43031,  
44110,  
45206,  
44622,  
44145,  
44622,  
44875,  
44230,  
45243,  
45429,  
44706,  
44278,  
45408,  
43734,  
43771,  
44624,  
43232,  
44053,  
43219,  
45872,  
45044,  
43351,  
44730,  
44730,  
44044,  
45044,  
45069,  
44110,  
45169,  
43219,  
45177,  
44137,  
44137,  
44137,  
43606,  
43028,  
43021,  
43211,  
43065,  
43065,  
43920,  
43016,  
43026,  
44307,  
44483,  
44484,  
44485,  
44485,  
44107,  
44107,  
44107,  
44107,  
43205,  
43845,  
44413,  
44325,  
43607,  
44134,  
44470.

45640,  
43606,  
45390,  
43614,  
43229,  
43608,  
44906,  
45211,  
43068,  
43055,  
43055,  
43055,  
43055,  
43055,  
45458,  
44052,  
44055,  
45426,  
43460,  
43228,  
44663,  
43050,  
44622,  
45320,  
45469,  
44135,  
43035,  
45885,  
45846,  
43130,  
45331,  
45619,  
45822,  
45690,  
44110,  
43078,  
45701,  
43832,  
43160,  
44662,  
44305,  
45331,  
43085,  
43205,  
44137,  
44104,  
45772,  
45613,  
43232,  
44095,  
45432,  
44683,  
44905,  
44012,  
44210,  
44074,  
43450,  
44067,  
45320,  
44224,  
43028,  
43517,  
43517,  
43611,  
44691,  
45152,  
43040,  
44004,  
44004,  
43204,  
43223,  
43604,  
45416,  
45662,  
44643,  
44643,  
44094,  
43231,  
-----

43224,  
44839,  
43935,  
44113,  
44805,  
45402,  
43212,  
44846,  
44646,  
44446,  
45807,  
43518,  
43518,  
44212,  
45011,  
45013,  
44903,  
43035,  
43015,  
43035,  
43021,  
43302,  
45219,  
43232,  
45767,  
43031,  
43001,  
44044,  
45807,  
44114,  
44114,  
43605,  
45312,  
44221,  
44707,  
45215,  
43440,  
44256,  
44312,  
45240,  
44401,  
45215,  
43935,  
43567,  
43551,  
43613,  
44125,  
45505,  
44817,  
45216,  
44035,  
44035,  
43081,  
45365,  
45322,  
45322,  
45323,  
44012,  
44133,  
43015,  
44319,  
44035,  
45697,  
43062,  
43213,  
44117,  
44112,  
45410,  
44123,  
44236,  
45631,  
45241,  
43085,  
43215,  
43540,  
43045,  
44118,  
44060,  
44408,

45014,  
43537,  
44837,  
45135,  
45014,  
45135,  
43107,  
43130,  
45669,  
45669,  
45669,  
44313,  
45365,  
45365,  
44662,  
43213,  
44077,  
45220,  
45406,  
43556,  
44114,  
44126,  
43205,  
44256,  
44256,  
43571,  
43147,  
43147,  
43147,  
44312,  
44119,  
45327,  
43616,  
43512,  
43227,  
45118,  
45118,  
44837,  
43116,  
43229,  
45778,  
45011,  
45015,  
45013,  
45013,  
44509,  
45120,  
45224,  
45120,  
45420,  
43973,  
44004,  
45385,  
44308,  
44052,  
45415,  
44120,  
44830,  
44260,  
44260,  
43160,  
43160,  
43215,  
44471,  
45219,  
45013,  
45840,  
44310,  
43223,  
45231,  
44074,  
44074,  
44313,  
44301,  
44256,  
45623,  
44138,  
44820,  
44270

```
44210,  
45895,  
45895,  
44224,  
44854,  
44138,  
44515,  
45324,  
45239,  
45215,  
44130,  
45867,  
44113,  
45373,  
44310,  
43103,  
44320,  
43229,  
44906,  
43214,  
45240,  
45403,  
44507,  
45342,  
44820,  
43202,  
43205,  
43213,  
43232,  
44105,  
43219,  
44041,  
44134,  
44134,  
44106,  
44113,  
44105,  
43701,  
44830,  
43227,  
44883,  
44304,  
44130,  
43068,  
45628,  
43074,  
44507,  
45405,  
45833,  
45358,  
...]
```

```
In [136]: # Run Census Search to retrieve data on all zip codes (2013 ACS5 Census)  
# See: https://github.com/CommerceDataService/census-wrapper for library documentation  
# See: https://gist.github.com/afhaque/60558290d6efd892351c4b64e5c01e9b for labels  
  
# set up a parameters dictionary  
ZIPCODE = []  
PAYANN = []  
  
for code in zcode:  
  
    # base url  
    query_url = f"https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:{code}&key={api_key}"  
  
    print(query_url)  
  
    #response = requests.get(base_url).json()  
  
    #Run requests to grab the JSON at the requested URL  
  
    response = requests.get(query_url)  
    print(response.status_code)  
  
    if response.status_code == 200:  
        jsonResponse = response.json()
```

```
#ZIPCODE
#print(EMPSZES)
#PAYANN
```



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44483&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44509&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43230&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44024&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44024&key=2af5575ace0b8385c92d40ea1d52f888>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44109&key=2a75575ace0b8385c92d40ea1d52f300>

https://api.census.gov/data/2016/zbp?get=z1PCODE,EMPSZES,PAYANN&for=z1pcode:44105&key=2af5575ace0b8385c92d40eald52f200

https://api.census.gov/data/2016/zbp?get=z1PCODE,EMPSZES,PAYANN&for=z1pcode:44/z1&key=2af5575ace0b8385c92d40ea1d52f200

https://api.census.gov/data/2016/zdp?get=z1PCODE,EMPSZES,PAYANN&for=z1pcode:44128&key=2af5575ace0b8385c92d40ea1d52f700

https://api.census.gov/data/2010/zbp:get-zipcode,EMP5LE5,FATAMN&for=-zipcode:45248&key=-zip5575ace0b8585c92d40ea1d521200

```
https://api.census.gov/data/2010/zdp:get-zip-code,ENR-SZES,PTATANDfor-zipcode:44011?key=5593ace6b585c52d40eaid521200
```

200

200

200

200

200

<https://doi.org/10.1080/09637480.2016.1172005>

200 [https://api.census.gov/data/2016/brb/get\\_ZIPCODE\\_FMPDZES\\_PAYANN?for=zipcode:44708&key=2af5575acc0b8285c03d40aa1d52f](https://api.census.gov/data/2016/brb/get_ZIPCODE_FMPDZES_PAYANN?for=zipcode:44708&key=2af5575acc0b8285c03d40aa1d52f)

200

<https://api.census.gov/data/2016/zhp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44610&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zhp?get=ZTPCODE,FMPS7ES,PAYANN&for=zipcode:45211&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45601&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44141&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44230&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43113&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43113&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44135&key=2a75575ace0b8385c92d40ea1d52f200>

https://api.census.gov/data/2016/zbp?get=z1PCODE,EMPSZES,PAYANN&for=z1pcode:44/08&key=2af5575ace0b8385c92d40eald52f200

https://api.census.gov/data/2010/zbp:get-zipcode,ENR5ZL5,FATAMN&for=-zipcode:45207&key=2af5575ace6b8585c52d40ea1d521200

200

200

200

200

200

200

[https://api.census.gov/data/2016/chr?\\_lang=en&\\_langcode=4506681&\\_langcode=2a55575&\\_langcode=9295-03+40+1+526](https://api.census.gov/data/2016/chr?_lang=en&_langcode=4506681&_langcode=2a55575&_langcode=9295-03+40+1+526)

200 <https://api.census.gov/data/2016/zhcten?get=ZIPCODE,EMPSIZE,PAYANN&for=zipcode:42102&key=3af5575ace0b8385c92d40ea1d52f>

https://api.census.gov/data/2016/zhp?get=7TPCODE,FMPS7FS,PAYANN&for=zipcode:45066&key=2af5575ace0b8385c92d40ea1d52f

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44720&key=2af5575ace0b8385c92d40ea1d52f>

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43739&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43762&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44108&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43551&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43610&key=2af5575ace0b8385c92d40ea1d52f888>

https://api.census.gov/data/2016/zbp?get=Z1PCODE,EMPSZES,PAYANN&for=z1pcode:45662&key=2af55/5ace0b8385c92d40eald52f200

https://api.census.gov/data/2016/z0p?get=z1PCODE,EMPSES,PAYANN&for=z1pcode:45/52&key=zaf3373ace008385c92d40ea1d52f700

```
https://api.census.gov/data/2010/zdp:get-zipcode,ENR,SZES,PTATANDfor-zipcode:44500&key-zai5575ace0b0505c52d40ea1d521
200
```

200

200

200

200

[illegible]

[https://api.census.gov/data/2016/chc?rest=ZIPCODE%20FIPS6756%20BAYANN&\\_lang=en&\\_csdcs=433338!&\\_scs=3a55575aa9b83295-03d40aa1d526](https://api.census.gov/data/2016/chc?rest=ZIPCODE%20FIPS6756%20BAYANN&_lang=en&_csdcs=433338!&_scs=3a55575aa9b83295-03d40aa1d526)

200

<https://api.census.gov/data/2016/zhr?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44647&key=3af5575a60b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zhp?get=7TPCODE,FMPS7FS,PAYANN&for=zipcode:45122&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45122&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45385&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45505&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43230&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45122&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45426&key=2a75575ace0b8385c92d40ea1d52f300>

https://api.census.gov/data/2016/zbp?get=Z1PCODE,EMPSZES,PAYANN&for=z1pcode:45224&key=2af5575ace0b8385c92d40eald52f200

https://api.census.gov/data/2010/z0p?get=z1PCODE,EMPSEZ,PAYANN&for=z1pcode:44143&key=zal5575ace0b8585c92d40ea1d521200

```
https://api.census.gov/data/2010/zsp/get-zip-code;E=5225;1ATAMM101--zipcode:45424&key=2af5575acc6b6565c52d40cd1d52f1200
```

200

200

200

200

200

200 <https://api.census.gov/data/2016/zhcten?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44206&key=2af5575ace0b8385c02d40ea1d52f>

200

<https://api.census.gov/data/2016/zhp?get=7TPCODE,EMPS7ES,PAYANN&for=zipcode:45232&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zhp?get=7TPCODE,FMPS7ES,PAYANN&for=zipcode:44109&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44301&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43402&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43219&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44076&key=2af5575ace0b8385c92d40ea1d52f>

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43055&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44444&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45359&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45244&key=2af5575ace0b8385c92d40ea1d52f388>

https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43556&key=2af5575ace0b8385c92d40ea1d52f200

https://api.census.gov/data/2016/zbp?get=Z1PCODE,EMPSZES,PAYANN&for=z1pcode:43113&key=2af5575ace0b8385c92d40ea1d52f700

https://api.census.gov/data/2016/zbp?get=z1PCODE,EMPSZES,PAYANN&for=z1pcode:45439&key=2af5575ace0b8385c92d40ea1d52f200

https://api.census.gov/data/2010/zdp/get-zipcode,EMP,SIZE,FAT,ANN&for=-zipcode.44121&key=2af5575ace008385c92d40ea1d521200

https://api.census.gov/data/2010/zbp/get-zipcode,EMPSZLS,PATANN&for=zipcode.44885&key=za1557face000505c32d40ea1d521200

[illegible]

200

200

200

200

200

200 <https://api.census.gov/data/2016/z/bp?get=ZIPCODE,EMPSIZE,PAYANN&for=zipcode:45011&key=2af5575ace0b8385c93d40ea1d52f>

200

<https://api.census.gov/data/2016/zhp?get=ZTPCODE,EMPSZES,PAYANN&for=zipcode:45690&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43920&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44667&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43078&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45891&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44647&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44310&key=2af5575ace0b8385c92d40ea1d52f888>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44095&key=2af5575ace0b8385c92d40ea1d52f200>

https://api.census.gov/data/2016/zbp?get=z1PCODE,EMPSZES,PAYANN&for=z1pcode:44505&key=2af5575ace0b8385c92d40ea1d52f200

https://api.census.gov/data/2016/zdp/get=ZIPCODE,EMP,SIZE,PAYANN&for=zipcode.44484&key=2af3573ace008385c932d40ea1d521200

```
https://api.census.gov/data/2010/zdp/get-zip-code?_lang=en&_format=json&_zipcode=44225&key=2af5575acc0b0505c92d40ca1d521200
```

200

200

200

200

|     |     |                                  |                                 |
|-----|-----|----------------------------------|---------------------------------|
| 111 | (1) | (1) + (2) = 77985 - 69875 = 8110 | (1) + (2) = 152150 - 152150 = 0 |
|-----|-----|----------------------------------|---------------------------------|

[https://api.census.gov/data/2016/zhcten/ztr/codes\\_5mpc75c\\_PAYANN8for\\_zincode:452158?\\_lang=en&\\_cs=3-65575---0b9385-02d40--1d526](https://api.census.gov/data/2016/zhcten/ztr/codes_5mpc75c_PAYANN8for_zincode:452158?_lang=en&_cs=3-65575---0b9385-02d40--1d526)

200 <https://api.census.gov/data/2016/z/bp?get=ZIPCODE,EMPSIZE,PAYANN&for=zipcode:45222&key=2af5575ace0b8385c93d40ea1d52f>

200 <https://api.census.gov/data/2016/zhp?get=ZTPCODE,FMPSZES,PAYANN&for=zipcode:44833&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45414&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43031&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44621&key=2af5575ace0b8385c92d40ea1d52f>

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44824&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44113&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43221&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44683&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45067&key=2af5575ace0b8385c92d40ea1d52f888>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43203&key=2a75575ace0b8385c92d40ea1d52f300>

https://api.census.gov/data/2016/zbp?get=z1PCODE,EMPSZES,PAYANN&for=z1pcode:43821&key=2af5575ace0b8385c92d40eald52f200

https://api.census.gov/data/2016/zbp?get=z1PCODE,EMPSZES,PAYANN&for=z1pcode:43000&key=2af5575ace008385c92d40ea1d52f200

https://api.census.gov/data/2010/zbp:get-zipcode,EMP5LE5,FATANN&for=-zipcode:43000&key=2af5575ace0b8585c92d40ea1d521200

200

```
https://api.census.gov/data/2019/zsp.get?_lang=en;zipCode=71814;stateKey=LA;stateFIPS=22;countyFIPS=009;tractFIPS=000100
```

200

200

200

200

[illegible]

200

[https://api.census.gov/data/2016/br?get=ZIPCODE5,EMPSZ5C,PAYANN5,sex,miscecd,AGE128!&\\_lang=en&\\_cs=2&\\_ts=2016-03-14T00:01:52.6](https://api.census.gov/data/2016/br?get=ZIPCODE5,EMPSZ5C,PAYANN5,sex,miscecd,AGE128!&_lang=en&_cs=2&_ts=2016-03-14T00:01:52.6)

200

[https://api.census.gov/data/2016/zhcten/get-ZIPCODE\\_FIPS750\\_PAYANN?for=zipcode:453388&key=3af5575ace0b8385c03d40ea1d52f](https://api.census.gov/data/2016/zhcten/get-ZIPCODE_FIPS750_PAYANN?for=zipcode:453388&key=3af5575ace0b8385c03d40ea1d52f)

<https://api.census.gov/data/2016/zhp?get=7TPCODE,FMPS7ES,PAYANN&for=zipcode:45381&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45322&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43844&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44805&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43023&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44514&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45069&key=2af5575ace0b8385c92d40ea1d52f300>

<https://api.census.gov/data/2016/zbp?get=Z1PCODE,EMPSZES,PAYANN&for=z1pcode:43/18&key=2af55/5ace0b8385c92d40eald52f200>

https://api.census.gov/data/2016/z0p?get=z1PCODE,EMPSES,PAYANN&for=z1pcode:43601&key=2af5575ace008385c92d40ea1d52f700

```
https://api.census.gov/data/2010/zbp/get-zip-code,zip=5225,pt=ATAMnafor+zipcode:44665akey=zip595acc0b8585c52d40cd1d52f  
200
```

200

200

200

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43351&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45859&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44132&key=2af5575ace0b8385c92d40ea1d52f888>

<https://api.census.gov/data/2016/zbp?get=Z1PCODE,EMPSZES,PAYANN&for=z1pcode:43078&key=2af5575ace0b8385c92d40ea1d52f200>

https://api.census.gov/data/2018/z0p?get=z1PCODE,EMPSTAT,PAYANN&for=z1pcode:43078&key=zaf3373ace008385c92d40ea1d52f700

```
https://api.census.gov/data/2010/zdp:get-zipcode,ENR SIZE,PTATAMNfor -zipcode:45000&key=2af5575ace6b8585c52d40ea1d521200
```

200

200

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45697&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43431&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45103&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45169&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45034&key=2af5575ace0b8385c92d40ea1d52f888>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43040&key=2a75575ace0b8385c92d40ea1d52f300>

https://api.census.gov/data/2016/zbp?get=z1PCODE,EMPSZES,PAYANN&for=z1pcode:45828&key=2af5575ace0b8385c92d40eald52f200

https://api.census.gov/data/2016/zbp?get=z1PCODE,EMPSZES,PAYANN&for=z1pcode:435/1&key=2af55/5ace008385c92d40ea1d52f700

https://api.census.gov/data/2010/zbp:get-zipcode,EMP5LE5,FATANN&for=-zipcode:43015&key=2af5575ace0b8585c92d40ea1d521200

200

[illegible]

200

200

200

[illegible]

200

[https://api.census.gov/data/2016/chr?\\_lang=en&\\_langcode=4414081&\\_lang=2a55575aa0b93295a03d40aa1d52c](https://api.census.gov/data/2016/chr?_lang=en&_langcode=4414081&_lang=2a55575aa0b93295a03d40aa1d52c)

200

[https://api.census.gov/data/2016/zhcten/get-ZIPCODE\\_FIPS750\\_PAYANN?for=zipcode:45040&key=3af5575ace0b8385c02d40ea1d52f](https://api.census.gov/data/2016/zhcten/get-ZIPCODE_FIPS750_PAYANN?for=zipcode:45040&key=3af5575ace0b8385c02d40ea1d52f)

<https://api.census.gov/data/2016/zhp?get=7TPCODE,FMPS7ES,PAYANN&for=zipcode:44011&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45817&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43065&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43035&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43718&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43718&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45044&key=2af5575ace0b8385c92d40ea1d52f300>

https://api.census.gov/data/2016/zbp?get=Z1PCODE,EMPSZES,PAYANN&for=z1pcode:45385&key=2af55/5ace0b8385c92d40eald52f200

https://api.census.gov/data/2016/z0p?get=z1PCODE,EMPSES,PAYANN&for=z1pcode:44320&key=2af5573ace008385c92d40ea1d52f700

[https://api.census.gov/data/2010/zbp/get-zip-code;\\_lang=en&\\_zip=57146&\\_key=zip](https://api.census.gov/data/2010/zbp/get-zip-code;_lang=en&_zip=57146&_key=zip)

200

200

200

200

[illegible]

200 <https://api.census.gov/data/2016/brb?get=ZIPCODE5,EMPSZES,PAYANN&for=zipcode:44707&key=2af5575ace0b8285e02d40aa1d526>

200

<https://api.census.gov/data/2016/zhp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44055&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zhp?get=7TPCODE,FMPS7ES,PAYANN&for=zipcode:45213&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45406&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43207&key=2af5575ace0b8385c92d40ea1d52f>



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

```

In [138]: ZIPCODE2 = []
PAYANN2 = []

for code in zcode2:

    # base url
    query_url = f"https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:{code}&key={api_key}"

    print(query_url)

    #response = requests.get(base_url).json()

    #Run requests to grab the JSON at the requested URL
    response2 = requests.get(query_url)
    print(response2.status_code)

    if response2.status_code == 200:
        jsonResponse2 = response2.json()
        ZIPCODE2.append(jsonResponse2[1][0])
        PAYANN2.append(jsonResponse2[1][2])

https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45404&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45207&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43537&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44115&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44663&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43207&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45414&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44320&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44709&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43604&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44130&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44502&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43609&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44484&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45237&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44883&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44134&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44111&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44102&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44134&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44134&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43613&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44906&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43920&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43227&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45505&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43229&key=2af5575ace0b8385c92d40ea1d52f
200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44134&key=2af5575ace0b8385c92d40ea1d52f
200

```

[illegible]

[illegible]



[illegible]

[illegible]

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43606&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44135&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45417&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43215&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43205&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43945&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43605&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43609&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44128&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45212&key=2af5575ace0b8385c92d40ea1d52f>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44119&key=2af5575ace0b8385c92d40ea1d52f888>

<https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44310&key=2a5575ace0b8385c92d40ea1d52f300>

https://api.census.gov/data/2016/zbp?get=Z1PCODE,EMPSZES,PAYANN&for=z1pcode:80203&key=2af55/5ace0b8385c92d40eald52f200

https://api.census.gov/data/2016/zbp?get=Z1PCODE,EMPSZES,PAYANN&for=z1pcode:43204&key=2af5575ace0b8385c92d40eald52f200

https://api.census.gov/data/2016/zbp?get=z1PCODE,EMPSZES,PAYANN&for=z1pcode:45506&key=2af5575ace008385c92d40eald52f200

https://api.census.gov/data/2016/zdp?get=z1PCODE,EMPSZES,PAYANN&for=z1pcode:43/z5&key=zatf55/5ace008385c92d40ea1d52f7200

https://api.census.gov/data/2018/z0p?get=z1PCODE,EMPSTAT,PAYANN&for=z1pcode:44108&key=2af3573ace008385c92d40ea1d52f700

https://api.census.gov/data/2010/z0p1?get=z1PCODE,EMPSEZ,PAYANN&for=z1pcode:43943&key=zal5575ace0b8585c92d40ea1d521200

https://api.census.gov/data/2010/zbp:get-zipcode,EMPSECS,FATANN&for=-zipcode:44128&key=2af5575ace0b8585c92d40ea1d521200

```
https://api.census.gov/data/2010/zdp:get-zip-code,ENR SZLS,1 ATANNdOr -zipcode:44700akey-zai 5575ace0b585c52d40eaid52i
200
```

[https://api.census.gov/data/2010/zbp/get-zip-code;\\_lang=en&\\_zip=571ATAMN&\\_zipcode=45226&key=\\_zip=571595dacc0b8585c52d4f0cd1d52f](https://api.census.gov/data/2010/zbp/get-zip-code;_lang=en&_zip=571ATAMN&_zipcode=45226&key=_zip=571595dacc0b8585c52d4f0cd1d52f)

```
https://api.census.gov/data/2010/20p/get=ZIP_CODE;EIN=5225;1ATAMM101=ZIPCODE:44765akey=ZIP55;5acc0b0505052d40cd1d52f1
200
```

[illegible]

200

200

200

200

200

200

200

200

200

200

200

200

200

[illegible]

200

[illegible]

```

https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43068&key=2af5575ace0b8385c92d40ea1d52f200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43605&key=2af5575ace0b8385c92d40ea1d52f200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45406&key=2af5575ace0b8385c92d40ea1d52f200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45011&key=2af5575ace0b8385c92d40ea1d52f200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45231&key=2af5575ace0b8385c92d40ea1d52f200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43232&key=2af5575ace0b8385c92d40ea1d52f200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45662&key=2af5575ace0b8385c92d40ea1d52f200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43302&key=2af5575ace0b8385c92d40ea1d52f200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43078&key=2af5575ace0b8385c92d40ea1d52f200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43302&key=2af5575ace0b8385c92d40ea1d52f200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:44052&key=2af5575ace0b8385c92d40ea1d52f200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:43337&key=2af5575ace0b8385c92d40ea1d52f200
https://api.census.gov/data/2016/zbp?get=ZIPCODE,EMPSZES,PAYANN&for=zipcode:45801&key=2af5575ace0b8385c92d40ea1d52f200

```

```

In [139]: # create dataframe
public_payfoll_by_ZIP_df = pd.DataFrame({"ZIPCODE": ZIPCODE,"Total Annual Payroll": PAYANN})
public_payfoll_by_ZIP_df.tail()

```

Out[139]:

|      | ZIPCODE | Total Annual Payroll |
|------|---------|----------------------|
| 3363 | 44287   | 26602                |
| 3364 | 43123   | 832214               |
| 3365 | 45813   | 6764                 |
| 3366 | 43762   | 38920                |
| 3367 | 45640   | 165770               |

```

In [140]: charter_payfoll_by_ZIP_df = pd.DataFrame({"ZIPCODE": ZIPCODE2,"Total Annual Payroll": PAYANN2})
charter_payfoll_by_ZIP_df.tail()

```

Out[140]:

|     | ZIPCODE | Total Annual Payroll |
|-----|---------|----------------------|
| 270 | 43078   | 299452               |
| 271 | 43302   | 743805               |
| 272 | 44052   | 124384               |
| 273 | 43337   | 0                    |
| 274 | 45801   | 685344               |

```

In [141]: csvpath = "Resources/Public_School_Data.csv"
raw_data = pd.read_csv(csvpath)
charter_schools_data = pd.read_csv("Resources/1617_CS_ACHIEVEMENT.csv")
charter_schools_data.columns

```

Out[141]: Index(['Building IRN', 'Building Name', 'District IRN', 'District Name', 'County', 'Region', 'Address', 'City and Zip Code', 'Phone #', 'Principal', 'Performance Index Score 2016-17', 'Performance Index Percent 2016-17', 'Letter Grade of Performance Index', 'Percent of Students Not Tested', 'Percent of Students Below', 'Percent of Students Basic', 'Percent of Students Proficient', 'Percent of Students Accelerated', 'Percent of Students Advanced', 'Percent of Students Advanced Plus', 'Gifted Performance Index Score 2016-17', 'Gifted Performance Index 2016-17', 'Percent of Gifted Students Not Tested', 'Percent of Gifted Students Below', 'Percent of Gifted Students Basic', 'Percent of Gifted Students Proficient', 'Percent of Gifted Students Accelerated', 'Percent of Gifted Students Advanced',

```
'Percent of Gifted Students Advanced Plus',
'Performance Index Score 2015-16', 'Performance Index Score 2014-15',
'Watermark', 'Unnamed: 32', 'Unnamed: 33'],
dtype='object')
```

```
In [142]: #Public and Charter School Datat
public_df = raw_data[["Building Name", "District Name", "County", "City, State, Zip Code", "Performance Index Score
charter_schools_df = charter_schools_data.loc[:, ['Building Name', 'District Name', 'County', 'City and Zip Code',
```

```
In [143]: charter_schools_df[["City", "State", "Zip Code"]] = charter_schools_df["City and Zip Code"].str.split(",",expand=True)
charter_schools_df.head()
```

Out[143]:

|   | Building Name                  | District Name                  | County     | City and Zip Code                | Performance Index Score |
|---|--------------------------------|--------------------------------|------------|----------------------------------|-------------------------|
| 0 | Pathway School of Discovery    | Pathway School of Discovery    | Montgomery | Dayton, OH, 45404-2123           | 79.49                   |
| 1 | Alliance Academy of Cincinnati | Alliance Academy of Cincinnati | Hamilton   | Cincinnati, OH, 45207-1644       | 64.585                  |
| 2 | Wildwood Environmental Academy | Wildwood Environmental Academy | Lucas      | Maumee, OH, 43537-1374           | 74.046                  |
| 3 | Ohio Connections Academy, Inc  | Ohio Connections Academy, Inc  | Cuyahoga   | Cleveland, OH, 44115-2229        | 77.619                  |
| 4 | Quaker Digital Academy         | New Philadelphia City          | Tuscarawas | New Philadelphia, OH, 44663-2150 | 65.397                  |

```
In [144]: charter_schools_df[["City", "State", "Zip Code"]] = charter_schools_df["City and Zip Code"].str.split(",",expand=True)
charter_schools_df[["Zip Code", "Code+4"]] = charter_schools_df["Zip Code"].str.split("-",expand=True).rename(columns={"Code+4": "City and Zip Code"})
charter_schools_df.head()
```

Out[144]:

|   | Building Name                  | District Name                  | County     | City and Zip Code                | Performance Index Score |
|---|--------------------------------|--------------------------------|------------|----------------------------------|-------------------------|
| 0 | Pathway School of Discovery    | Pathway School of Discovery    | Montgomery | Dayton, OH, 45404-2123           | 79.49                   |
| 1 | Alliance Academy of Cincinnati | Alliance Academy of Cincinnati | Hamilton   | Cincinnati, OH, 45207-1644       | 64.585                  |
| 2 | Wildwood Environmental Academy | Wildwood Environmental Academy | Lucas      | Maumee, OH, 43537-1374           | 74.046                  |
| 3 | Ohio Connections Academy, Inc  | Ohio Connections Academy, Inc  | Cuyahoga   | Cleveland, OH, 44115-2229        | 77.619                  |
| 4 | Quaker Digital Academy         | New Philadelphia City          | Tuscarawas | New Philadelphia, OH, 44663-2150 | 65.397                  |

```
In [145]: public_df[["City", "State", "Zip Code"]] = public_df["City, State, Zip Code"].str.split(",",expand=True)
public_df[["Zip Code", "Code+4"]] = public_df["Zip Code"].str.split("-",expand=True)
public_df.head()
```

C:\Users\jnovic\AppData\Local\Continuum\anaconda3\lib\site-packages\pandas\core\frame.py:3137: SettingWithCopyWarning: A value is trying to be set on a copy of a slice from a DataFrame.  
Try using .loc[row\_indexer,col\_indexer] = value instead

See the caveats in the documentation: <http://pandas.pydata.org/pandas-docs/stable/indexing.html#indexing-view-versus-copy>  
self[k1] = value[k2]

Out[145]:

|   | Building Name             | District Name        | County | City, State, Zip Code      | Performance Index Score 2015-16 | City       |
|---|---------------------------|----------------------|--------|----------------------------|---------------------------------|------------|
| 0 | Ada Elementary School     | Ada Exempted Village | Hardin | Ada, OH, 45810-1013        | 96.125                          | Ada        |
| 1 | Ada High School           | Ada Exempted Village | Hardin | Ada, OH, 45810-1013        | 91.667                          | Ada        |
| 2 | Sandusky Middle School    | Sandusky City        | Erie   | Sandusky, OH, 44870-2616   | 62.772                          | Sandusky   |
| 3 | Meigs Primary School      | Meigs Local          | Meigs  | Middleport, OH, 45760-9717 | NC                              | Middleport |
| 4 | Meigs Intermediate School | Meigs Local          | Meigs  | Middleport, OH, 45760-9717 | 70.394                          | Middleport |

```
In [146]: public_df.to_csv("Resources/Public_ZipCode.csv", index=False, header=True)
```

```
In [147]: public_payroll= public_payroll_by_ZIP_df.rename(columns={"ZIPCODE": "Zip Code"})
public_payroll.head()
```

Out[147]:

|   | Zip Code | Total Annual Payroll |
|---|----------|----------------------|
| 0 | 45810    | 77875                |
| 1 | 45810    | 77875                |
| 2 | 44870    | 894118               |
| 3 | 45760    | 9171                 |

|   |       |      |
|---|-------|------|
| 4 | 45760 | 9171 |
|---|-------|------|

In [148]: `public_df.head()`

Out[148]:

|   | Building Name             | District Name        | County | City, State, Zip Code      | Performance Index Score 2015-16 | City       |
|---|---------------------------|----------------------|--------|----------------------------|---------------------------------|------------|
| 0 | Ada Elementary School     | Ada Exempted Village | Hardin | Ada, OH, 45810-1013        | 96.125                          | Ada        |
| 1 | Ada High School           | Ada Exempted Village | Hardin | Ada, OH, 45810-1013        | 91.667                          | Ada        |
| 2 | Sandusky Middle School    | Sandusky City        | Erie   | Sandusky, OH, 44870-2616   | 62.772                          | Sandusky   |
| 3 | Meigs Primary School      | Meigs Local          | Meigs  | Middleport, OH, 45760-9717 | NC                              | Middleport |
| 4 | Meigs Intermediate School | Meigs Local          | Meigs  | Middleport, OH, 45760-9717 | 70.394                          | Middleport |

In [149]: `public_df["Zip Code"] = public_df["Zip Code"].astype(int)`  
`public_payroll["Zip Code"] = public_payroll["Zip Code"].astype(int)`  
`public_payroll.dtypes`

C:\Users\jnovic\AppData\Local\Continuum\anaconda3\lib\site-packages\ipykernel\_launcher.py:1: SettingWithCopyWarning  
A value is trying to be set on a copy of a slice from a DataFrame.  
Try using `.loc[row_indexer,col_indexer] = value` instead

See the caveats in the documentation: <http://pandas.pydata.org/pandas-docs/stable/indexing.html#indexing-view-versus-copy>  
"""Entry point for launching an IPython kernel.

Out[149]: Zip Code int32  
Total Annual Payroll object  
dtype: object

In [150]: `#Problems Merging`  
`public_pay_df = public_df.merge(public_payroll, on="Zip Code")`  
`public_pay_df.head()`

Out[150]:

|   | Building Name          | District Name        | County | City, State, Zip Code    | Performance Index Score 2015-16 | City     | S |
|---|------------------------|----------------------|--------|--------------------------|---------------------------------|----------|---|
| 0 | Ada Elementary School  | Ada Exempted Village | Hardin | Ada, OH, 45810-1013      | 96.125                          | Ada      | O |
| 1 | Ada Elementary School  | Ada Exempted Village | Hardin | Ada, OH, 45810-1013      | 96.125                          | Ada      | O |
| 2 | Ada High School        | Ada Exempted Village | Hardin | Ada, OH, 45810-1013      | 91.667                          | Ada      | O |
| 3 | Ada High School        | Ada Exempted Village | Hardin | Ada, OH, 45810-1013      | 91.667                          | Ada      | O |
| 4 | Sandusky Middle School | Sandusky City        | Erie   | Sandusky, OH, 44870-2616 | 62.772                          | Sandusky | O |

In [151]: `charter_payroll = charter_payroll_by_ZIP_df.rename(columns={"ZIPCODE": "Zip Code"})`  
`charter_schools_df["Zip Code"] = charter_schools_df["Zip Code"].astype(int)`  
`charter_payroll["Zip Code"] = charter_payroll["Zip Code"].astype(int)`  
`charter_pay_df = pd.merge(charter_schools_df, charter_payroll, on="Zip Code")`  
`charter_pay_df.head()`

Out[151]:

|   | Building Name                  | District Name                  | County     | City and Zip Code          | Performance Index Score 2015-16 |
|---|--------------------------------|--------------------------------|------------|----------------------------|---------------------------------|
| 0 | Pathway School of Discovery    | Pathway School of Discovery    | Montgomery | Dayton, OH, 45404-2123     | 79.49                           |
| 1 | Pathway School of Discovery    | Pathway School of Discovery    | Montgomery | Dayton, OH, 45404-2123     | 79.49                           |
| 2 | Pathway School of Discovery    | Pathway School of Discovery    | Montgomery | Dayton, OH, 45404-2123     | 79.49                           |
| 3 | Alliance Academy of Cincinnati | Alliance Academy of Cincinnati | Hamilton   | Cincinnati, OH, 45207-1644 | 64.585                          |
| 4 | Alliance Academy of Cincinnati | Alliance Academy of Cincinnati | Hamilton   | Cincinnati, OH, 45207-1644 | 64.585                          |

In [152]: `public_df = public_df.loc[public_df["Performance Index Score 2015-16"] != "NC"]`  
`charter_schools_df = charter_schools_df.loc[charter_schools_df["Performance Index Score 2016-17"] != "NC"]`  
`public_df["Performance Index Score 2015-16"] = public_df["Performance Index Score 2015-16"].astype(float)`  
`charter_schools_df["Performance Index Score 2016-17"] = charter_schools_df["Performance Index Score 2016-17"].astype(float)`

In [153]: `#Ethnicity/Race Data`  
`csvpath2 = "Resources/Ethnicity.csv"`  
`new_data_race = pd.read_csv(csvpath2)`

```
raw_data_race = pd.read_csv(csvpath1)
ethnic_df = pd.read_csv(csvpath2)
```

```
C:\Users\jnovic\AppData\Local\Continuum\anaconda3\lib\site-packages\IPython\core\interactiveshell.py:2785: DtypeWarning:
on on import or set low_memory=False.
interactivity=interactivity, compiler=compiler, result=result)
```

In [ ]:

```
In [154]: ### Replace the >95
race_df = raw_data_race[["Building Name", "Student Group", "% of Total Enrollment"]]
race_df = race_df.replace(to_replace = "NC", value = 0)
race_df["% of Total Enrollment"] = race_df["% of Total Enrollment"].astype(str)
race_df["% of Total Enrollment"] = race_df["% of Total Enrollment"].replace(to_replace = r'>95', value = 95.1, regex=True)
race_df["% of Total Enrollment"] = race_df["% of Total Enrollment"].astype(float)
```

```
In [155]: df_white = race_df.loc[race_df["Student Group"] == "White", :]
df_latinx = race_df.loc[race_df["Student Group"] == "Hispanic", :]
df_black = race_df.loc[race_df["Student Group"] == "Black", :]
```

In [156]: race\_df.head(30)

Out[156]:

|    | Building Name             | Student Group                     | % of Total Enrollment |
|----|---------------------------|-----------------------------------|-----------------------|
| 0  | Ada Elementary School     | American Indian or Alaskan Native | 0.0                   |
| 1  | Ada Elementary School     | Asian or Pacific Islander         | 0.0                   |
| 2  | Ada Elementary School     | Black                             | 3.4                   |
| 3  | Ada Elementary School     | Hispanic                          | 0.0                   |
| 4  | Ada Elementary School     | Multiracial                       | 0.0                   |
| 5  | Ada Elementary School     | White                             | 92.5                  |
| 6  | Ada High School           | American Indian or Alaskan Native | 0.0                   |
| 7  | Ada High School           | Asian or Pacific Islander         | 0.0                   |
| 8  | Ada High School           | Black                             | 0.0                   |
| 9  | Ada High School           | Hispanic                          | 0.0                   |
| 10 | Ada High School           | Multiracial                       | 0.0                   |
| 11 | Ada High School           | White                             | 94.1                  |
| 12 | Sandusky Middle School    | American Indian or Alaskan Native | 0.0                   |
| 13 | Sandusky Middle School    | Asian or Pacific Islander         | 0.0                   |
| 14 | Sandusky Middle School    | Black                             | 38.1                  |
| 15 | Sandusky Middle School    | Hispanic                          | 4.7                   |
| 16 | Sandusky Middle School    | Multiracial                       | 18.8                  |
| 17 | Sandusky Middle School    | White                             | 37.9                  |
| 18 | Meigs Primary School      | American Indian or Alaskan Native | 0.0                   |
| 19 | Meigs Primary School      | Asian or Pacific Islander         | 0.0                   |
| 20 | Meigs Primary School      | Black                             | 0.0                   |
| 21 | Meigs Primary School      | Hispanic                          | 0.0                   |
| 22 | Meigs Primary School      | Multiracial                       | 3.7                   |
| 23 | Meigs Primary School      | White                             | 94.5                  |
| 24 | Meigs Intermediate School | American Indian or Alaskan Native | 0.0                   |
| 25 | Meigs Intermediate School | Asian or Pacific Islander         | 0.0                   |
| 26 | Meigs Intermediate School | Black                             | 0.0                   |
| 27 | Meigs Intermediate School | Hispanic                          | 0.0                   |
| 28 | Meigs Intermediate School | Multiracial                       | 0.0                   |
| 29 | Meigs Intermediate School | White                             | 95.1                  |

In [157]: public\_race\_white = df\_white.merge(public\_df, on="Building Name")



```

In [157]: public_race_latinx = df_latinx.merge(public_df, on="Building Name")
public_race_black = df_black.merge(public_df, on="Building Name")

charterwhite_df = pd.merge(charter_schools_df, df_white, on="Building Name")
charterblack_df = pd.merge(charter_schools_df, df_black, on="Building Name")
charterhispanic_df = pd.merge(charter_schools_df, df_latinx, on="Building Name")

```

```

In [158]: #Econ Data
csvpath3 = "Resources/Economic.csv"
raw_data_econ = pd.read_csv(csvpath3)
raw_data_econ.head()

```

Out[158]:

|   | Building<br>IRN | Building<br>Name       | District<br>IRN | District<br>Name     | County | Region   | Open/Closed<br>Status as of<br>9/12/2017 | School<br>Type    | Student Group    | Read 3rd<br>Grade<br>2016-<br>2017 %<br>Proficient<br>or above | ... | Geometry<br>End of<br>Course<br>2016-<br>2017 %<br>Proficient<br>or Above |
|---|-----------------|------------------------|-----------------|----------------------|--------|----------|--|-------------------|------------------|--|-----|---|
| 0 | 59              | Ada Elementary School  | 45187           | Ada Exempted Village | Hardin | Region 6 | Open                                     | Elementary School | Disadvantaged    | 77.4   | ... | NC  |
| 1 | 59              | Ada Elementary School  | 45187           | Ada Exempted Village | Hardin | Region 6 | Open                                     | Elementary School | NonDisadvantaged | 92.3   | ... | NC  |
| 2 | 67              | Ada High School        | 45187           | Ada Exempted Village | Hardin | Region 6 | Open                                     | High School       | Disadvantaged    | NC   | ... | 56.0  |
| 3 | 67              | Ada High School        | 45187           | Ada Exempted Village | Hardin | Region 6 | Open                                     | High School       | NonDisadvantaged | NC   | ... | 62.9  |
| 4 | 83              | Sandusky Middle School | 44743           | Sandusky City        | Erie   | Region 2 | Open                                     | Middle School     | Disadvantaged    | NC   | ... | NC  |

5 rows × 41 columns

```

In [159]: econ_df = raw_data_econ[["Building Name", "Student Group", "% of Total Enrollment"]]
econ_df = econ_df.replace(to_replace = "NC", value = 0)
econ_df["% of Total Enrollment"] = econ_df["% of Total Enrollment"].astype(str)
econ_df["% of Total Enrollment"] = econ_df["% of Total Enrollment"].replace(to_replace = r'>95', value = 95.1, regex=True)
econ_df["% of Total Enrollment"] = econ_df["% of Total Enrollment"].astype(float)

```

```

In [160]: df_poor = econ_df.loc[econ_df["Student Group"] == "Disadvantaged", :]
df_notpoor = econ_df.loc[econ_df["Student Group"] == "NonDisadvantaged", :]

```

```

In [161]: public_poor_df = df_poor.merge(public_df, on="Building Name")
public_notpoor_df = df_notpoor.merge(public_df, on="Building Name")

charter_disadvantaged = pd.merge(charter_schools_df, df_poor, on="Building Name")
charter_nondisadvantaged = pd.merge(charter_schools_df, df_notpoor, on="Building Name")

```

```

In [162]: #Data to analyze
public_black_cuya = public_race_black.loc[public_race_black["County"] == "Cuyahoga", :]
public_white_cuya = public_race_white.loc[public_race_white["County"] == "Cuyahoga", :]
public_latinx_cuya = public_race_latinx.loc[public_race_latinx["County"] == "Cuyahoga", :]
public_poor_cuya = public_poor_df.loc[public_poor_df["County"] == "Cuyahoga", :]
public_notpoor_cuya = public_notpoor_df.loc[public_notpoor_df["County"] == "Cuyahoga", :]

charter_nondisadv_cuyahoga = charter_nondisadvantaged.loc[charter_nondisadvantaged["County"] == "Cuyahoga"]
charter_disadv_cuyahoga = charter_disadvantaged.loc[charter_disadvantaged["County"] == "Cuyahoga"]
charterwhite_cuyahoga = charterwhite_df.loc[charterwhite_df["County"] == "Cuyahoga"]
charterblack_cuyahoga = charterblack_df.loc[charterblack_df["County"] == "Cuyahoga"]
charterhispanic_cuyahoga = charterhispanic_df.loc[charterhispanic_df["County"] == "Cuyahoga"]

public_pay_cuya = public_pay_df.loc[public_pay_df["County"] == "Cuyahoga"]
charter_nav_cuya = charter_nav_df.loc[charter_nav_df["County"] == "Cuyahoga"]

```

```
In [163]: public_black_frank = public_race_black.loc[public_race_black["County"] == "Franklin", :]
public_white_frank = public_race_white.loc[public_race_white["County"] == "Franklin", :]
public_latinx_frank = public_race_latinx.loc[public_race_latinx["County"] == "Franklin", :]
public_poor_frank = public_poor_df.loc[public_poor_df["County"] == "Franklin", :]
public_notpoor_frank = public_notpoor_df.loc[public_notpoor_df["County"] == "Franklin", :]

charter_nondisadv_franklin = charter_nondisadvantaged.loc[charter_nondisadvantaged["County"] == "Franklin"]
charter_disadv_franklin = charter_disadvantaged.loc[charter_disadvantaged["County"] == "Franklin"]
charterwhite_franklin = charterwhite_df.loc[charterwhite_df["County"] == "Franklin"]
charterblack_franklin = charterblack_df.loc[charterblack_df["County"] == "Franklin"]
charterhispanic_franklin = charterhispanic_df.loc[charterhispanic_df["County"] == "Franklin"]

public_pay_franklin = public_pay_df.loc[public_pay_df["County"] == "Franklin"]
charter_pay_franklin = charter_pay_df.loc[charter_pay_df["County"] == "Franklin"]
```

```
In [164]: public_black_ham = public_race_black.loc[public_race_black["County"] == "Hamilton", :]
public_white_ham = public_race_white.loc[public_race_white["County"] == "Hamilton", :]
public_latinx_ham = public_race_latinx.loc[public_race_latinx["County"] == "Hamilton", :]
public_poor_ham = public_poor_df.loc[public_poor_df["County"] == "Hamilton", :]
public_notpoor_ham = public_notpoor_df.loc[public_notpoor_df["County"] == "Hamilton", :]

charter_nondisadv_hamilton = charter_nondisadvantaged.loc[charter_nondisadvantaged["County"] == "Hamilton"]
charter_disadv_hamilton = charter_disadvantaged.loc[charter_disadvantaged["County"] == "Hamilton"]
charterwhite_hamilton = charterwhite_df.loc[charterwhite_df["County"] == "Hamilton"]
charterblack_hamilton = charterblack_df.loc[charterblack_df["County"] == "Hamilton"]
charterhispanic_hamilton = charterhispanic_df.loc[charterhispanic_df["County"] == "Hamilton"]

public_pay_ham = public_pay_df.loc[public_pay_df["County"] == "Hamilton"]
charter_pay_ham = charter_pay_df.loc[charter_pay_df["County"] == "Hamilton"]
```

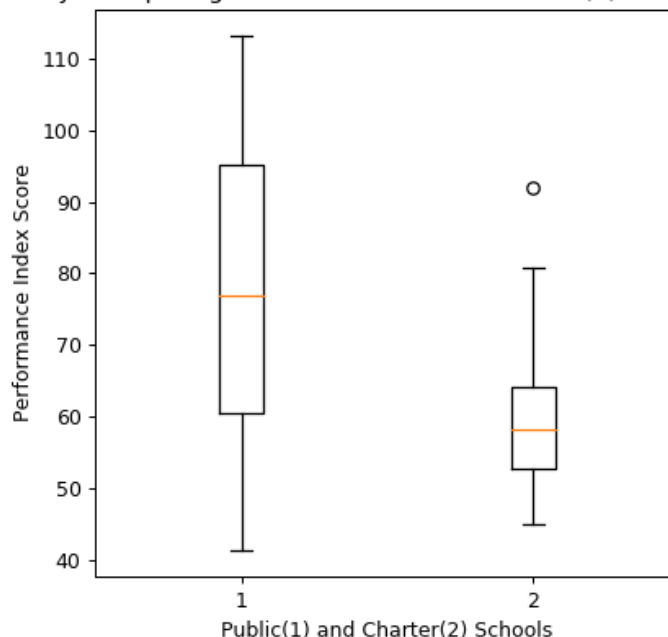
```
In [232]: #t-test
public_df_ham = public_df.loc[public_df["County"] == "Hamilton"]
charter_df_ham = charter_schools_df.loc[charter_schools_df["County"] == "Hamilton"]
s1=public_df_ham["Performance Index Score 2015-16"]
s2=charter_df_ham["Performance Index Score 2016-17"]

plt.boxplot([s1, s2])
plt.title("Hamilton County: Comparing Performance between Public(1) and Charter(2) Schools")
plt.xlabel("Public(1) and Charter(2) Schools")
plt.ylabel("Performance Index Score")
(t_stat, p) = stats.ttest_ind(s1, s2, equal_var=False)
print(t_stat)
print(p)

fig_pub_ham = plt.gcf()
```

<IPython.core.display.Javascript object>

Hamilton County: Comparing Performance between Public(1) and Charter(2) Schools



```
5.4672700433479715
6.595916637671241e-06
```

```
In [233]: plt.tight_layout()
fig_pub_ham .savefig("Images/Hamilton_County_Public_Charter.png")
plt.show()
```

```
In [234]: # Public Black Regression

x_ham_blk = public_black_ham["% of Total Enrollment"]
y_ham_blk = public_black_ham["Performance Index Score 2015-16"]

(hbk_slope, hbk_int, hbk_c_r, hbk_p, hbk_std_err) = stats.linregress(x_ham_blk, y_ham_blk)
fit = hbk_slope * x_ham_blk + hbk_int

fig, ax = plt.subplots()

fig.suptitle("Relationship Between Ethnicity and Public School Performance Score (Black Students)", fontsize=16, fo

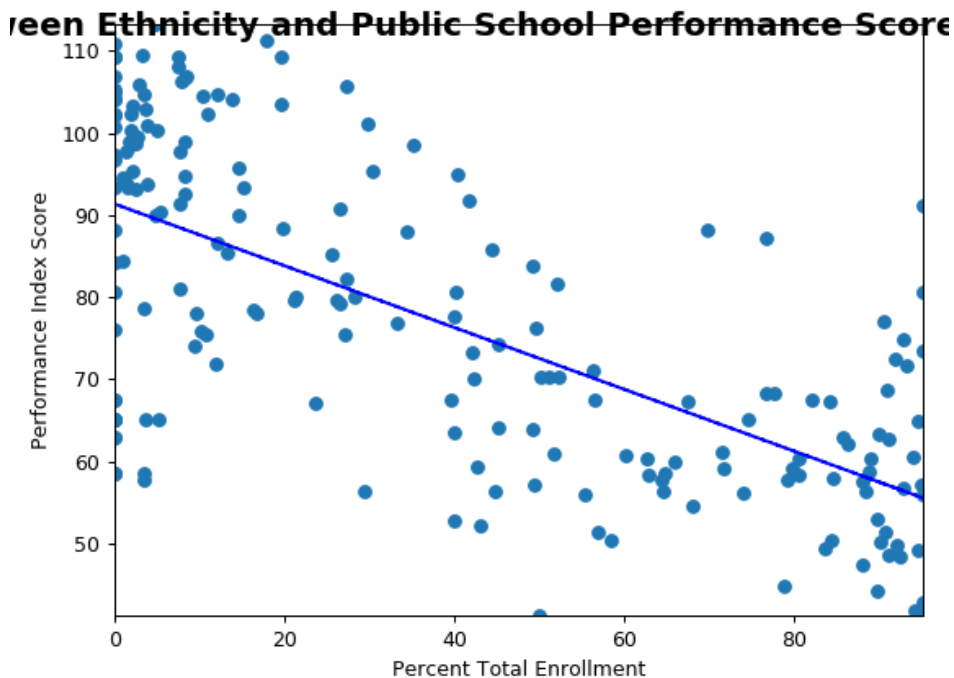
ax.set_xlim(min(x_ham_blk), max(x_ham_blk))
ax.set_ylim(min(y_ham_blk), max(y_ham_blk))

ax.set_xlabel("Percent Total Enrollment")
ax.set_ylabel("Performance Index Score")

ax.plot(x_ham_blk, y_ham_blk, linewidth=0, marker='o')
ax.plot(x_ham_blk, fit, 'b--')

fig_pubblk_ham = plt.gcf()
plt.show()
```

<IPython.core.display.Javascript object>



```
In [235]: plt.tight_layout()
fig_pubblk_ham .savefig("Images/Hamilton_Count_Public_Blkc.png")
plt.show()
```

```
In [236]: #Charter School Black Performance

x_ham_blk2 = charterblack_hamilton["% of Total Enrollment"]
y_ham_blk2 = charterblack_hamilton["Performance Index Score 2016-17"]

(hbk2_slope, hbk2_int, hbk2_c_r, hbk2_p, hbk2_std_err) = stats.linregress(x_ham_blk2, y_ham_blk2)
fit = hbk2_slope * x_ham_blk2 + hbk2_int

fig, ax = plt.subplots()
```

```
fig.suptitle("Relationship Between Ethnicity and Charter School Performance Score (Black Students)", fontsize=16, f
ax.set_xlim(min(x_ham_blk2), max(x_ham_blk2))
ax.set_ylim(min(y_ham_blk2), max(y_ham_blk2))

ax.set_xlabel("Percent Total Enrollment")
ax.set_ylabel("Performance Index Score")

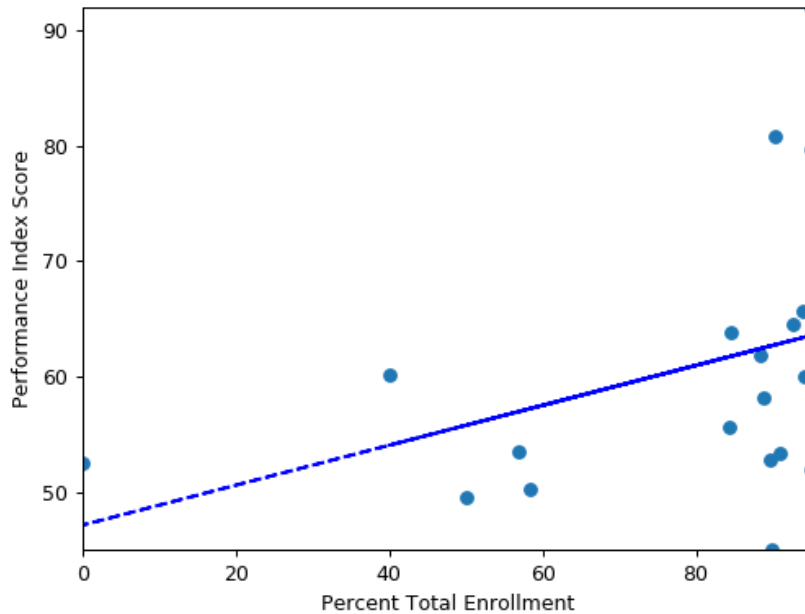
ax.plot(x_ham_blk2, y_ham_blk2, linewidth=0, marker='o')
ax.plot(x_ham_blk2, fit, 'b--')

fig_charblk_ham = plt.gcf()

plt.show()
```

<IPython.core.display.Javascript object>

## een Ethnicity and Charter School Performance Scor



◀

```
In [170]: plt.tight_layout()
fig_charblk_ham.savefig("Images/Hamilton_Count_Charter_Blkc.png")
plt.show()
```

<Figure size 432x288 with 0 Axes>

```
In [237]: # Public Hispanix Regression

x_ham_lat = public_latinx_ham["% of Total Enrollment"]
y_ham_lat = public_latinx_ham["Performance Index Score 2015-16"]

(hlk_slope, hlk_int, hlk_c_r, hlk_p, hlk_std_err) = stats.linregress(x_ham_lat, y_ham_lat)
fit = hlk_slope * x_ham_lat + hlk_int

fig, ax = plt.subplots()

fig.suptitle("Relationship Between Ethnicity and Public School Performance Score (Latino/a Students)", fontsize=16,

ax.set_xlim(min(x_ham_lat), max(x_ham_lat))
ax.set_ylim(min(y_ham_lat), max(y_ham_lat))

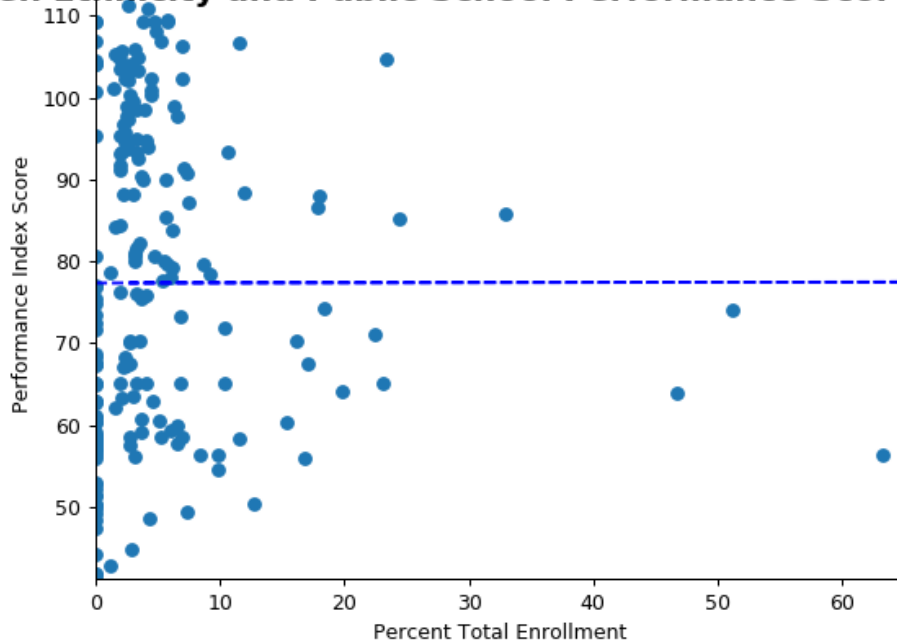
ax.set_xlabel("Percent Total Enrollment")
ax.set_ylabel("Performance Index Score")

ax.plot(x_ham_lat, y_ham_lat, linewidth=0, marker='o')
ax.plot(x_ham_lat, fit, 'b--')
fig_publat_ham = plt.gcf()

plt.show()
```

<IPython.core.display.Javascript object>

## an Ethnicity and Public School Performance Score



```
In [238]: plt.tight_layout()
fig_publat_ham.savefig("Images/Hamilton_Count_Public_latino.png")
plt.show()
```

```
In [239]: #Charter School Hispanic Performance

x_ham_lat2 = charterhispanic_hamilton["% of Total Enrollment"]
y_ham_lat2 = charterhispanic_hamilton["Performance Index Score 2016-17"]

(hlk2_slope, hlk2_int, hlk2_c_r, hlk2_p, hlk2_std_err) = stats.linregress(x_ham_lat2, y_ham_lat2)
fit = hlk2_slope * x_ham_lat2 + hlk2_int

fig, ax = plt.subplots()

fig.suptitle("Relationship Between Ethnicity and Charter School Performance Score (Latino/a Students)", fontsize=16)

ax.set_xlim(min(x_ham_lat2), max(x_ham_lat2))
ax.set_ylim(min(y_ham_lat2), max(y_ham_lat2))

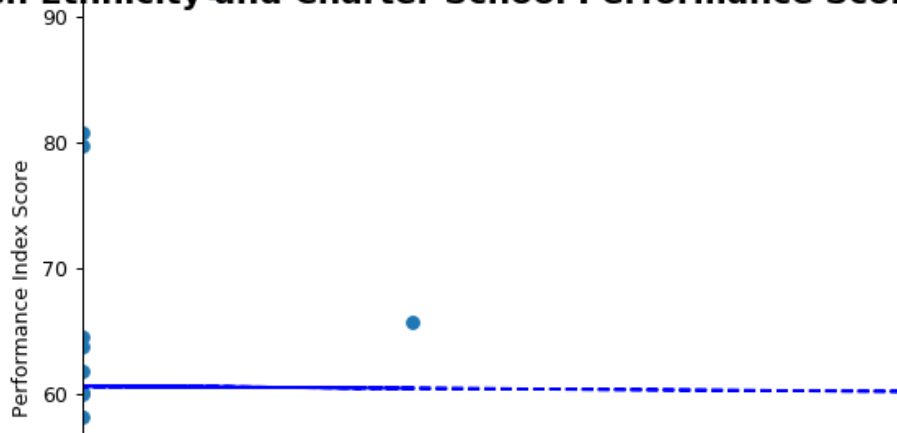
ax.set_xlabel("Percent Total Enrollment")
ax.set_ylabel("Performance Index Score")

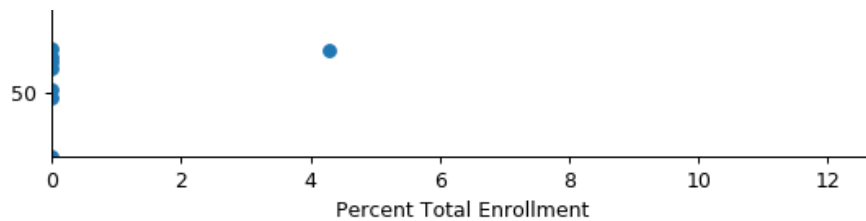
ax.plot(x_ham_lat2, y_ham_lat2, linewidth=0, marker='o')
ax.plot(x_ham_lat2, fit, 'b--')
fig_charlat_ham = plt.gcf()

plt.show()
```

<IPython.core.display.Javascript object>

## an Ethnicity and Charter School Performance Score





```
In [240]: plt.tight_layout()
fig_charlat_ham.savefig("Images/Hamilton_Count_Charter_latino.png")
plt.show()
```

```
In [241]: # Public White Regression

x_ham_wh = public_white_ham["% of Total Enrollment"]
y_ham_wh = public_white_ham["Performance Index Score 2015-16"]

(hwk_slope, hwk_int, hwk_c_r, hwk_p, hwk_std_err) = stats.linregress(x_ham_wh, y_ham_wh)
fit = hwk_slope * x_ham_wh + hwk_int

fig, ax = plt.subplots()

fig.suptitle("Relationship Between Ethnicity and Public School Performance Score (White Students)", fontsize=16, fo

ax.set_xlim(min(x_ham_wh), max(x_ham_wh))
ax.set_ylim(min(y_ham_wh), max(y_ham_wh))

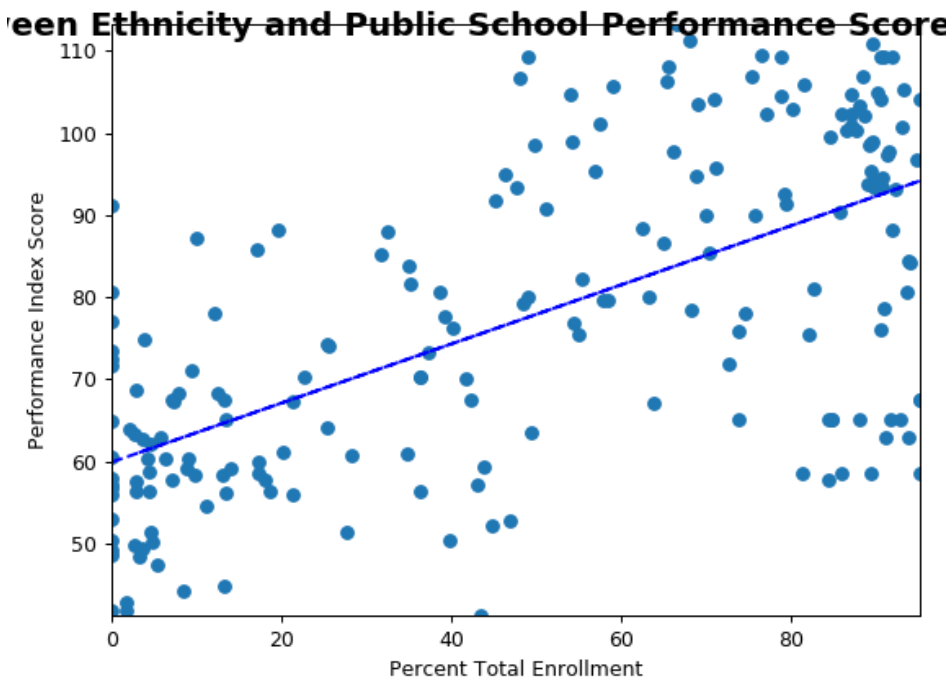
ax.set_xlabel("Percent Total Enrollment")
ax.set_ylabel("Performance Index Score")

ax.plot(x_ham_wh, y_ham_wh, linewidth=0, marker='o')
ax.plot(x_ham_wh, fit, 'b--')

fig_pubwh_ham = plt.gcf()

plt.show()
```

<IPython.core.display.Javascript object>



```
In [242]: plt.tight_layout()
fig_pubwh_ham.savefig("Images/Hamilton_Count_Public_white.png")
plt.show()
```

```
In [243]: x_ham_wh2 = charterwhite_hamilton["% of Total Enrollment"]
```

```

y_ham_wh2 = charterwhite_hamilton["Performance Index Score 2016-17"]

(hwk2_slope, hwk2_int, hwk2_c_r, hwk2_p, hwk2_std_err) = stats.linregress(x_ham_wh2, y_ham_wh2)
fit = hwk2_slope * x_ham_wh2 + hwk2_int

fig, ax = plt.subplots()

fig.suptitle("Relationship Between Ethnicity and Charter School Performance Score (White Students)", fontsize=16, fontweight='bold')

ax.set_xlim(min(x_ham_wh2), max(x_ham_wh2))
ax.set_ylim(min(y_ham_wh2), max(y_ham_wh2))

ax.set_xlabel("Percent Total Enrollment")
ax.set_ylabel("Performance Index Score")

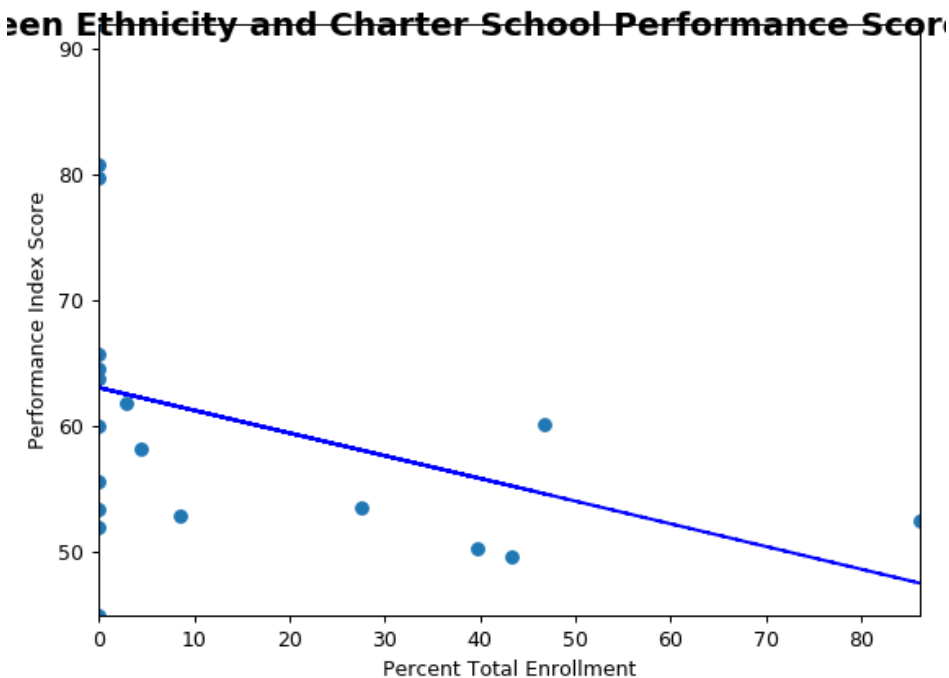
ax.plot(x_ham_wh2, y_ham_wh2, linewidth=0, marker='o')
ax.plot(x_ham_wh2, fit, 'b--')

fig_charwh_ham = plt.gcf()

plt.show()

```

<IPython.core.display.Javascript object>



```

In [244]: plt.tight_layout()
fig_charwh_ham.savefig("Images/Hamilton_Count_Charter_white.png")
plt.show()

```

```

In [245]: x_charter_blk_all = charterblack_df["% of Total Enrollment"]
y_charter_blk_all = charterblack_df["Performance Index Score 2016-17"]

(cbc_slope, cbc_int, cbc_c_r, cbc_p, cbc_std_err) = stats.linregress(x_charter_blk_all, y_charter_blk_all)
fit = cbc_slope * x_charter_blk_all + cbc_int

fig, ax = plt.subplots()

fig.suptitle("Relationship Between Ethnicity and All Ohio Charter Schools' Performance Score (Black Students)", fontweight='bold')

ax.set_xlim(min(x_charter_blk_all), max(x_charter_blk_all))
ax.set_ylim(min(y_charter_blk_all), max(y_charter_blk_all))

ax.set_xlabel("Percent Total Enrollment")
ax.set_ylabel("Performance Index Score")

ax.plot(x_charter_blk_all, y_charter_blk_all, linewidth=0, marker='o')
ax.plot(x_charter_blk_all, fit, 'b--')

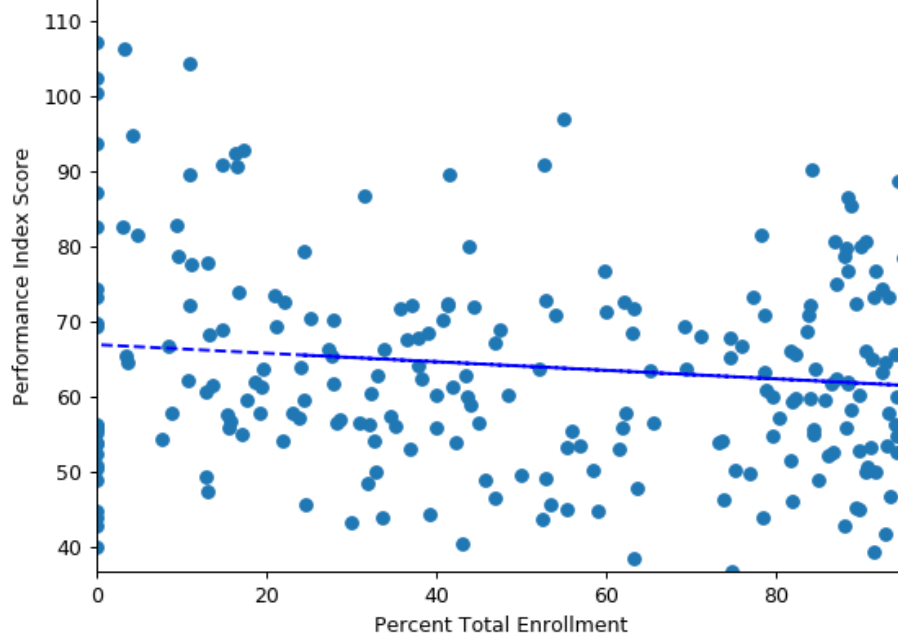
fig_allblk_ham = plt.gcf()

```

```
plt.show()
```

<IPython.core.display.Javascript object>

## ~~thnicity and All Ohio Charter Schools' Performance~~



```
In [246]: plt.tight_layout()
fig_allblk_ham.savefig("Images/All_OHIO_Charter_black.png")
plt.show()
```

```
In [247]: x_public_ham_poor =public_poor_ham["% of Total Enrollment"]
y_public_ham_poor =public_poor_ham["Performance Index Score 2015-16"]

(hpp_slope, hpp_int, hpp_c_r, hpp_p, hpp_std_err) = stats.linregress(x_public_ham_poor, y_public_ham_poor)
fit = hpp_slope * x_public_ham_poor + hpp_int

fig, ax = plt.subplots()

fig.suptitle("Relationship Between Poverty and Public Schools' Performance Score", fontsize=16, fontweight="bold")

ax.set_xlim(min(x_public_ham_poor), max(x_public_ham_poor))
ax.set_ylim(min(y_public_ham_poor), max(y_public_ham_poor))

ax.set_xlabel("Percent Total Enrollment")
ax.set_ylabel("Performance Index Score")

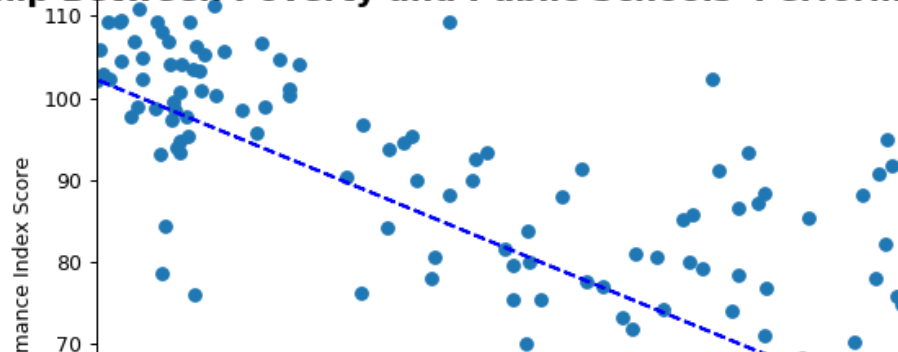
ax.plot(x_public_ham_poor,y_public_ham_poor, linewidth=0, marker='o')
ax.plot(x_public_ham_poor, fit, 'b--')

fig_pubpoor_ham = plt.gcf()

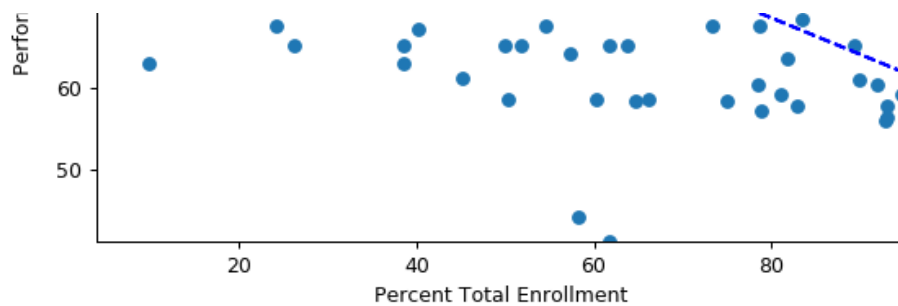
plt.show()
```

<IPython.core.display.Javascript object>

## ~~hip Between Poverty and Public Schools' Performa~~







```
In [248]: plt.tight_layout()
fig_pubpoor_ham.savefig("Images/Hamilton_County_Poverty_Public.png")
plt.show()
```

```
In [249]: x_charter_ham_poor = charter_disadv_hamilton["% of Total Enrollment"]
y_charter_ham_poor = charter_disadv_hamilton["Performance Index Score 2016-17"]

(hpc_slope, hpc_int, hpc_c_r, hpc_p, hpc_std_err) = stats.linregress(x_charter_ham_poor, y_charter_ham_poor)
fit = hpc_slope * x_charter_ham_poor + hpc_int

fig, ax = plt.subplots()

fig.suptitle("Relationship Between Poverty and Charter Schools' Performance Score", fontsize=16, fontweight="bold")

ax.set_xlim(min(x_charter_ham_poor), max(x_charter_ham_poor))
ax.set_ylim(min(y_charter_ham_poor), max(y_charter_ham_poor))

ax.set_xlabel("Percent Total Enrollment")
ax.set_ylabel("Performance Index Score")

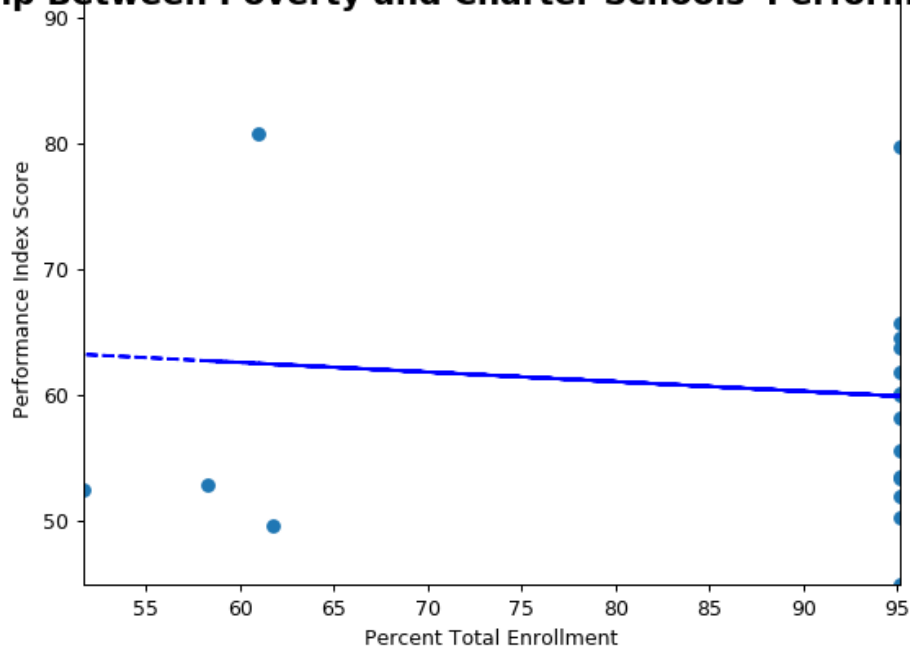
ax.plot(x_charter_ham_poor, y_charter_ham_poor, linewidth=0, marker='o')
ax.plot(x_charter_ham_poor, fit, 'b--')

fig_charpoor_ham = plt.gcf()

plt.show()
```

<IPython.core.display.Javascript object>

### Relationship Between Poverty and Charter Schools' Performance



```
In [250]: plt.tight_layout()
fig_charpoor_ham.savefig("Images/Hamilton_County_Poverty_Charter.png")
plt.show()
```

```
In [195]: #charter_pay_ham = charter_pay_ham.Loc[charter_pay_ham["Performance Index Score 2016-17"] != "NC"]
charter_pay_ham["Total Annual Payroll"]=charter_pay_ham["Total Annual Payroll"].astype(float)
charter_pay_ham["Performance Index Score 2016-17"]=charter_pay_ham["Performance Index Score 2016-17"].astype(float)

public_pay_ham["Total Annual Payroll"]=public_pay_ham["Total Annual Payroll"].astype(float)
public_pay_ham["Performance Index Score 2015-16"]=public_pay_ham["Performance Index Score 2015-16"].astype(float)
```

```
In [251]: x_pub_ham_pay =public_pay_ham["Total Annual Payroll"]
y_public_ham_pay =public_pay_ham["Performance Index Score 2015-16"]

(pph_slope, pph_int, pph_c_r, pph_p, pph_std_err) = stats.linregress(x_pub_ham_pay,y_public_ham_pay)
fit = pph_slope * x_pub_ham_pay + pph_int

fig, ax = plt.subplots()

fig.suptitle("Relationship Between Annual Payroll in Zip Code and Public Schools' Performance Score", fontsize=16,

ax.set_xlim(min(x_pub_ham_pay), max(x_pub_ham_pay))
ax.set_ylim(min(y_public_ham_pay), max(y_public_ham_pay))

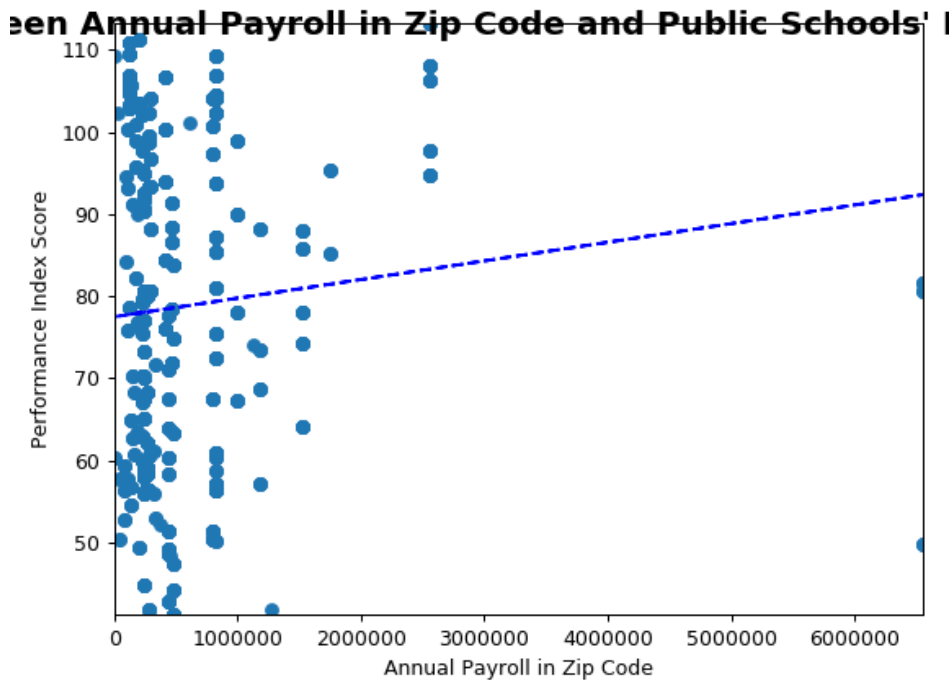
ax.set_xlabel("Annual Payroll in Zip Code")
ax.set_ylabel("Performance Index Score")

ax.plot(x_pub_ham_pay,y_public_ham_pay, linewidth=0, marker='o')
ax.plot(x_pub_ham_pay, fit, 'b--')

fig_pubpay_ham = plt.gcf()

plt.show()
```

<IPython.core.display.Javascript object>



```
In [252]: plt.tight_layout()
fig_pubpay_ham.savefig("Images/HamiltonCountyPayrollPublic.png")
plt.show()
```

```
In [253]: x_chart_ham_pay =charter_pay_ham["Total Annual Payroll"]
y_chart_ham_pay =charter_pay_ham["Performance Index Score 2016-17"]

(cph_slope, cph_int, cph_c_r, cph_p, cph_std_err) = stats.linregress(x_chart_ham_pay ,y_chart_ham_pay)
fit = cph_slope * x_chart_ham_pay + cph_int

fig, ax = plt.subplots()

fig.suptitle("Relationship Between Annual Payroll in Zip Code and Charter Schools' Performance Score", fontsize=16,

ax.set_xlim(min(x_chart_ham_pay), max(x_chart_ham_pay))
```

```

ax.set_xlim(min(x_chart_ham_pay), max(x_chart_ham_pay))
ax.set_ylim(min(y_chart_ham_pay), max(y_chart_ham_pay))

ax.set_xlabel("Annual Payroll in Zip Code")
ax.set_ylabel("Performance Index Score")

ax.plot(x_chart_ham_pay, y_chart_ham_pay, linewidth=0, marker='o')
ax.plot(x_chart_ham_pay, fit, 'b--')

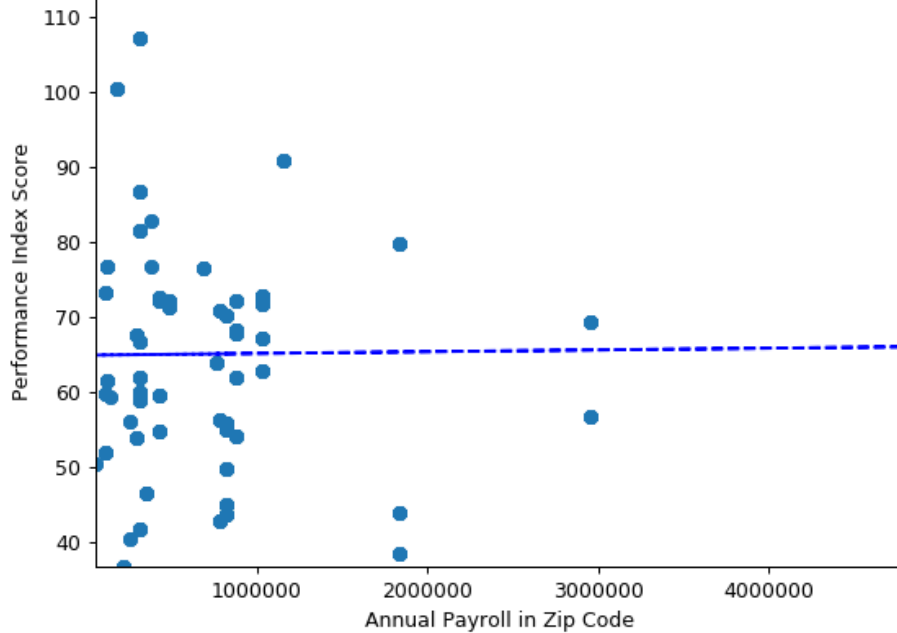
fig_charpay_ham = plt.gcf()

plt.show()

```

<IPython.core.display.Javascript object>

## en Annual Payroll in Zip Code and Charter Schools'



```

In [254]: plt.tight_layout()
fig_charpay_ham.savefig("Images/Hamilton_County_Payroll_Charter.png")
plt.show()

```

```

In [211]: print("Black Public CR and P "+ str(hbk_c_r) +" , " + str(hbk_p))
print("Black Charter CR and P "+ str(hbk2_c_r) +" , " + str(hbk2_p))
print("Latino Public CR and P "+ str(hlk_c_r) +" , " + str(hlk_p))
print("Latino Charter CR and P "+ str(hlk2_c_r) +" , " + str(hlk2_p))
print("White Public CR and P "+ str(hwk_c_r) +" , " + str(hwk_p))
print("White Charter CR and P "+ str(hwk2_c_r) +" , " + str(hwk2_p))
print("Poor Public CR and P "+ str(hpp_c_r) +" , " + str(hpp_p))
print("Poor Charter CR and P "+ str(hpc_c_r) +" , " + str(hpc_p))
print("Pay Public CR and P "+ str(pph_c_r) +" , " + str(pph_p))
print("Pay Charter CR and P "+ str(cph_c_r) +" , " + str(cph_p))

```

Black Public CR and P -0.6756783613598567 , 9.316875847059938e-28  
 Black Charter CR and P 0.36429478539890475 , 0.12518393728600985  
 Latino Public CR and P 0.0008860608496832295 , 0.9901152275572891  
 Latino Charter CR and P -0.008705018247513071 , 0.9717857232038377  
 White Public CR and P 0.6415554281664281 , 2.3623439886552994e-24  
 White Charter CR and P -0.35960916688771594 , 0.1304826299672473  
 Poor Public CR and P -0.7662946633760132 , 1.6257311401638353e-39  
 Poor Charter CR and P -0.0996420105342553 , 0.6848484921517045  
 Pay Public CR and P 0.08760922137020619 , 0.005858677694300336  
 Pay Charter CR and P 0.009301861523096162 , 0.7970257131337074

In [221]:

|                       |     |
|-----------------------|-----|
| Building Name         | 368 |
| Student Group         | 368 |
| % of Total Enrollment | 368 |
| District Name         | 368 |
| County                | 368 |
| City                  | 368 |

|                                 |     |
|---------------------------------|-----|
| City, State, Zip Code           | 368 |
| Performance Index Score 2015-16 | 368 |
| City                            | 368 |
| State                           | 368 |
| Zip Code                        | 368 |
| Code+4                          | 367 |
| dtype: int64                    |     |
| Building Name                   | 368 |
| Student Group                   | 368 |
| % of Total Enrollment           | 368 |
| District Name                   | 368 |
| County                          | 368 |
| City, State, Zip Code           | 368 |
| Performance Index Score 2015-16 | 368 |
| City                            | 368 |
| State                           | 368 |
| Zip Code                        | 368 |
| Code+4                          | 367 |
| dtype: int64                    |     |
| Building Name                   | 368 |
| Student Group                   | 368 |
| % of Total Enrollment           | 368 |
| District Name                   | 368 |
| County                          | 368 |
| City, State, Zip Code           | 368 |
| Performance Index Score 2015-16 | 368 |
| City                            | 368 |
| State                           | 368 |
| Zip Code                        | 368 |
| Code+4                          | 367 |
| dtype: int64                    |     |
| Building Name                   | 368 |
| Student Group                   | 368 |
| % of Total Enrollment           | 368 |
| District Name                   | 368 |
| County                          | 368 |
| City, State, Zip Code           | 368 |
| Performance Index Score 2015-16 | 368 |
| City                            | 368 |
| State                           | 368 |
| Zip Code                        | 368 |
| Code+4                          | 367 |
| dtype: int64                    |     |
| Building Name                   | 368 |
| Student Group                   | 368 |
| % of Total Enrollment           | 368 |
| District Name                   | 368 |
| County                          | 368 |
| City, State, Zip Code           | 368 |
| Performance Index Score 2015-16 | 368 |
| City                            | 368 |
| State                           | 368 |
| Zip Code                        | 368 |
| Code+4                          | 367 |
| dtype: int64                    |     |
| Building Name                   | 63  |
| District Name                   | 63  |
| County                          | 63  |
| City and Zip Code               | 63  |
| Performance Index Score 2016-17 | 63  |
| City                            | 63  |
| State                           | 63  |
| Zip Code                        | 63  |
| Code+4                          | 63  |
| Student Group                   | 63  |
| % of Total Enrollment           | 63  |
| dtype: int64                    |     |
| Building Name                   | 63  |
| District Name                   | 63  |
| County                          | 63  |
| City and Zip Code               | 63  |
| Performance Index Score 2016-17 | 63  |
| City                            | 63  |
| State                           | 63  |
| Zip Code                        | 63  |
| Code+4                          | 63  |
| Student Group                   | 63  |
| % of Total Enrollment           | 63  |
| dtype: int64                    |     |

```

Building Name          63
District Name          63
County                 63
City and Zip Code      63
Performance Index Score 2016-17 63
City                   63
State                  63
Zip Code               63
Code+4                 63
Student Group          63
% of Total Enrollment  63
dtype: int64
Building Name          63
District Name          63
County                 63
City and Zip Code      63
Performance Index Score 2016-17 63
City                   63
State                  63
Zip Code               63
Code+4                 63
Student Group          63
% of Total Enrollment  63
dtype: int64
Building Name          63
District Name          63
County                 63
City and Zip Code      63
Performance Index Score 2016-17 63
City                   63
State                  63
Zip Code               63
Code+4                 63
Student Group          63
% of Total Enrollment  63
dtype: int64
Building Name          2995
District Name          2995
County                 2995
City, State, Zip Code  2995
Performance Index Score 2015-16 2995
City                   2995
State                  2995
Zip Code               2995
Code+4                 2963
Total Annual Payroll   2995
dtype: int64
Building Name          718
District Name          718
County                 718
City and Zip Code      718
Performance Index Score 2016-17 718
City                   718
State                  718
Zip Code               718
Code+4                 718
Total Annual Payroll   718
dtype: int64

```

In [227]: charter\_nondisadv\_hamilton.count()

```

Out[227]: Building Name          19
District Name          19
County                 19
City and Zip Code      19
Performance Index Score 2016-17 19
City                   19
State                  19
Zip Code               19
Code+4                 19
Student Group          19
% of Total Enrollment  19
dtype: int64

```

In [ ]:









































































































